Jameco Part number 148726

Vitrax Plus^{tm} Controller

REAL-TIME CLOCK MODULE

Document #940504-CK

REAL-TIME CALENDAR CLOCK MANUAL

INSTALLATION PROCEDURE
HARDWARE ARRANGEMENT
SOFTWARE SUPPORT

CONTROL TECHNOLOGY

Vitrax Plus is a trademark of Control Technology, Box 4605, Mtn View CA 94040. The Vitrax Plus Controller design and associated documentation in this manual are copyrighted by Control Technology. All rights are reserved. Duplication in any form is prohibited without the written consent of Control Technology.

Jameco Part number 148726

REAL TIME CALENDAR CLOCK

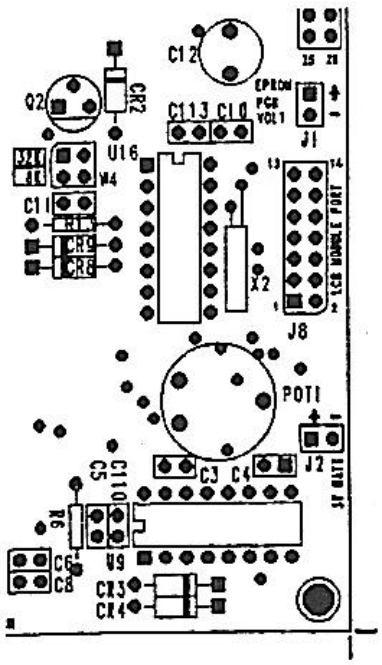
OVERVIEW: The Real-Time Calendar Clock Module adds date and time capability to the **Vitrax Plus Controller.** It can also produce an independent source of interrupt pulses at programmable rates. The module consists of 11 components that are installed at the middle-right side of the pc board (plus battery holder).

ASSEMBLY INSTRUCTIONS: Refer to the Parts List and the Component Arrangement Diagram. The recommended assembly sequence is shown below.

ASSEMBLY SEQUENCE

COMPONENT	DESCRIPTION	COMMENTS
X2	32.768 KHz xtal	See p/c board screened outline
R13	10K resistor	1/8-watt
CR8,9	1N270 diodes	Alternates are 1N5817,8
C10	15 pf disc ceramic capc	
C11	100 pf disc ceramic capc	
C113	0.1 uf monolithic capc	
Z16	16-pin socket	
C12	5-30 pf trimmer capc	
W11	2-pin jumper	Write enable/disable jumper
Q2	2N7000 xstr	
U16	58274	
JB1	Jumper block	Place on W11 to enable write
-	Battery holder	Attaches to J2

COMPONENT ARRANGEMENT DIAGRAM



Jameco Part number 148726

The 58274 clock device incorporates 6 basic functions that are described below:

* Calibration of the clock frequency
* Setting date including leap year compensation
* Setting time including hour format (AM/PM vs 24-hour)
* Starting/stopping clock
* Reading clock
* Setting and enabling interrupt pulses

CLOCK ARCHITECTURE: U16, a 58274 CMOS integrated timekeeping device, contains 16 4-bit registers; 14 are used as date/time cascading counters and 2 are used for control, interrupt and status. Refer to the architecture table below. Registers are grouped to handle both date and time functions. The range of date capability includes month, day and day-of-the-week for up to 10 years with leap year compensation. Timekeeping functions cover hours (12 or 24-hour format), minutes, seconds and 1/10th second. The functional description in the attached data sheet explains the operation of each register. A brief register outline follows:

REGISTERS: The register set is shown in the following table. Register 0 at I/O address 160 performs different functions in Read and Write mode as indicated in the table. Also, keep in mind that bit 1 in Register 0 (highlighted) determines whether Register 15 at I/O address 175 functions as a clock Setting Register or as an Interrupt Control Register. Similarly, bit 3 selects either normal mode (clock and interrupt functions) or test (calibration) mode. Both the clock and the interrupt functions embodied in Register 15 (highlighted) are shown in the table:

CALIBRATION OF THE CLOCK FREQUENCY: The clock device is crystal controlled for accuracy. A trimmer capacitor, C12, can be used for precise adjustment, but the oscillator circuit needs to be buffered and gated to interrupt pin 13 to avoid loading the circuit during frequency measurement. Before attaching a frequency counter to pin 13, execute steps 1 and 2 to condition the oscillator signal:

1. Ensure that the clock and interrupt timers are halted - and that test mode and interrupt registers are selected: Write 15 (decimal) to control register 0: **OUT 160,15**

2. Gate the oscillator to the interrupt pin; ensure that no interrupts are programmed: Write 0 to register 15: **OUT 175,0**

3. Timing is halted. Attach a frequency counter to pin 13 and set the oscillator by adjusting C12. Optimal reading is 32,768 KHz.

4. Escape from the test to clock setting mode. Write 5 to the control register 0 **OUT 160,5**

ARCHITECTURE OF CLOCK REGISTERS

REG	ADDRESS DEC	ADDRESS HEX	FUNCTION	ACCESS	DATA BITS DB3	DB2	DB1	DB0	
0	160	A0	Data changed flag	R	x				1=Clock data changed
			Interrupt flag	R				x	1=Interrupt timeout
			Normal / test mode	W	x				0=Normal 1=Test
			Clock start/stop	W		x			0=Run 1=Stop
			Set clock/intr (Reg 15)	***W***			x		***0=Clk Set 1=Intr Set***
			Interrupt start/stop	W				x	0=Run 1=Stop
1	161	A1	Tenths second	R	x	x	x	x	
2	162	A2	Units seconds	R/W	x	x	x	x	
3	163	A3	Tens seconds	R/W	-	x	x	x	
4	164	A4	Units minutes	R/W	x	x	x	x	
5	165	A5	Tens minutes	R/W	-	x	x	x	
6	166	A6	Units hours	R/W	x	x	x	x	
7	167	A7	Tens hours	R/W	-	-	x	x	
8	168	A8	Units days	R/W	x	x	x	x	
9	169	A9	Tens days	R/W	-	-	x	x	
10	170	AA	Units month	R/W	x	x	x	x	
1	171	AB	Tens month	R/W	-	-	-	x	
12	172	AC	Units years	R/W	x	x	x	x	
13	173	AD	Tens years	R/W	x	x	x	x	
14	174	AE	Day of Week	R/W	-	x	x	x	
15	***175***	***AF***	***Clk set if Reg 0,DB1=0***	***R/W***	x	x			0=leap year **
							x		0=AM 1=PM **
								x	0=12hr 1=24hr **
15	***175***	***AF***	***Intr set if Reg 0,DB1=1***	***2R/W***	x	0	0	0	No interrupt
					0/1*	0	0	1	0.1 second
					0/1*	0	1	0	0.5 second
					0/1*	0	1	1	1 second
					0/1*	1	0	0	5 second
					0/1*	1	1	1	10 second
					0/1*	1	0	0	30 second
					0/1*	1	1	1	60 second

* With Register 15 in Interrupt Mode (DB1=1 in Register 0), DB3 in Register 15 determines if the pulse on pin 13 is single or repeating: DB3=0 for single pulse or DB3=1 for repeating interrupts pulses.

** The 12/24-hour format must be set by a separate command from the AM/PM indicator even though both items are in the same register. In other words, the 12/24-hours format bit (DB0) and the AM-PM/24-hour bit (DB1) cannot be set in the same write operation.

SETTING THE DATE including leap year compensation: ***Always set the date before the time.*** Use the OUT command to set the date in Registers 8 to 14. Example, Wednesday, May 29, 1987:

1. Select the clock mode with the clock stopped. Write 5 to control register 0: OUT 160,5

2. Program the date into registers 8 through 14 - followed by a leap year compensation factor in register 15. The factor is equal to the number of years since the last leap year. Use a factor of zero if the year being programmed is a leap year. At this point, disregard bits DB0 and DB1 in register 15. The following table shows the date for Wednesday, May 29, 1987:

Jameco Part number 148726

REG	ADDR	DATA	DB3	DB2	DB1	DB0	
11	161	0	0	0	0	0	Tens month
10	170	5	0	1	0	1	Units month
9	169	2	0	0	1	0	Tens days
8	168	9	1	0	0	1	Units days
13	173	8	1	0	0	0	Tens years
12	172	7	0	1	1	1	Units years
14	174	4	0	1	0	0	Day of week
15	175	12	1	1	x	x	Years since leap year

SETTING THE TIME: Use the OUT command to set time ***after setting the date.*** Begin the process by ensuring that the clock mode (vs test or interrupt mode) is selected.

1. Select the clock mode with the clock stopped. Write 5 to control register 0: **OUT 160,5**

2. Select either 12-or 24-hour. Write a 0 for 12-hour or a 1 for 24-hour. IMPORTANT: see the note in step four about AM-PM setting if 12-hour format is selected. Be sure to carry the leap year compensation bits in DB3 and DB2 programmed in step 2 under Setting The Date.

	REG	ADDR	DATA	DB3	DB2	DB1	DB0	
	15	175	0	x	x	x	0	For 12-hour format
or	15	175	1	x	x	x	1	For 24-hour format

12/24-hour format

3. Program the time into registers 1 through 7. (If 12-hour format is selected in step 2, be certain to complete step 4). For example: (8:47:00 PM). (Set DATE before setting TIME):

REG	ADDR	DATA	DB3	DB2	DB1	DB0		
7	167	0	0	0	0	0	Tens hrs	OUT 167,0
6	166	8	1	0	0	0	Units hrs	OUT 166,8
5	165	4	0	1	0	0	Tens min	OUT 165,4
4	164	7	0	1	1	1	Units min	OUT 164,7
3	163	0	0	0	0	0	Tens sec	OUT 163,0
2	162	0	0	0	0	0	Units sec	OUT 162,0

(Continue to step 4 to select AM=PM, if 12/24 hour format has been selected (step 2 above)

4. If 12-hour format was selected in step 2, complete this step; otherwise, proceed to step 5. Continuing the example from step 3, complete the setting of PM in the AM-PM bit

	REG	ADDR	DATA	DB3	DB2	DB1	DB0	
	15	175	0	x	x	0	x	For AM in 12-hr format
or	15	175	1	x	x	1	x	For PM in 12-hour format

AM/PM for 12=hr format

The 12/24-hour format must be set by a separate command from the AM/PM indicator even though both items are in the same register. In other words, the 12/24-hours format bit (DB0) and the AM-PM/24-hour bit (DB1) cannot be set in the same write operation. Be sure to carry the leap year compensation bits in DB3 and DB2 when setting the 12/24-hour format and the AM/PM indicator.

5. Start the clock, typically **OUT 160,1** where DB2=0 with interrupts off from the clock device.

STARTING/STOPPING THE CLOCK: The clock timekeeping operation is started and stopped by writing to register 0. The clock can be started with interrupts from the clock device disabled or enabled using the same command:

To start the clock with interrupts disabled:	**OUT 160,1**
To start the clock with interrupts enabled:	**OUT 160,0**
To stop the clock with interrupts disabled:	**OUT 160,5**
To stop the clock with interrupts enabled:	**OUT 160,4**

READING THE CLOCK: Reading the time and date is similar to the set procedure except the INPut command is used and the clock is usually read "on the fly", A status bit, DB3, is provided in Register 0 to inform the processor if the time has changed. This flag is cleared only by a read of Register 0. When INPutting data from the clock register, a byte is received. Only the lower 4-bits (D0-D3) pertain to the clock. Since all data lines are pulled up, the upper 4-bits can be discarded by subtracting 240 from the INPutted value.

SETTING & USING INTERRUPT PULSES: The internal interrupt timer generates interrupts at intervals that are programmed into the interrupt register 15 at I/O address 175. Keep in mind that this register address is shared with clock function - and its access depends on the settings in the control register (register 0 at I/O address 160, particularly DB0 and DB1). A single interrupt after delay or repeated interrupts may be selected based of the state of DB3 in register 15. Program this bit with a 0 for a single pulse or a 1 for repeated pulses.

1. To select the interrupt register with interrupt timing stopped while clock (date/time) functions continue, write a 3 to register 0 at I/O address 160. **OUT 160,3**

2. Write an interrupt control word to register 15 at I/O address 175. Determine the appropriate word from the following table for both timing interval and single or repeating interrupt pulses:

REG	**ADDRESS**		**DATA BITS**				
	DEC	**HEX**	**DB3**	**DB2**	**DB1**	**DB0**	
15	175	AF	X	0	0	0	No interrupt
			0/1*	0	0	1	0.1 second
			0/1*	0	1	0	0.5 second
			0/1*	0	1	1	1 second
			0/1*	1	0	0	5 seconds
			0/1*	1	0	1	10 seconds
			0/1*	1	1	0	30 seconds
			0/1*	1	1	1	60 seconds

* When Register 15 is in Interrupt Mode (by DB1=1 in Register 0), DB3 in Register 15 determines if the output interrupt pulse on pin 13 is a single or repeating interrupt: Set DB3=0 for a single pulse or DB3=1 for repeating pulses.

3. Write a 0 or 2 to control register 0 at I/O address 160. Interrupt timing begins. A 0 gates the clock setting register onto the bus. Writing 2 leaves the interrupt register selected. Normal timekeeping remains unaffected.

Jameco Part number 148726

SUPPORTING SOFTWARE: A BASIC routine is provided below to set and read the time and date. The program is menu oriented, but it is easily modified for use in application programs. Here are some explanatory notes about the code:

1. Lines 7002-7023: A menu is provided to select one of four functions with the selection stored in Z0.

2. Lines 8505-12: The date is entered from the console as D$ with some rudimentary error checking for an 8-character format in lines 8510 and 8512.

3. Line 8514: Day-of-the-week is stored as D0.

4. Lines 8516-20: These OUTput statements enter the date into the clock registers.

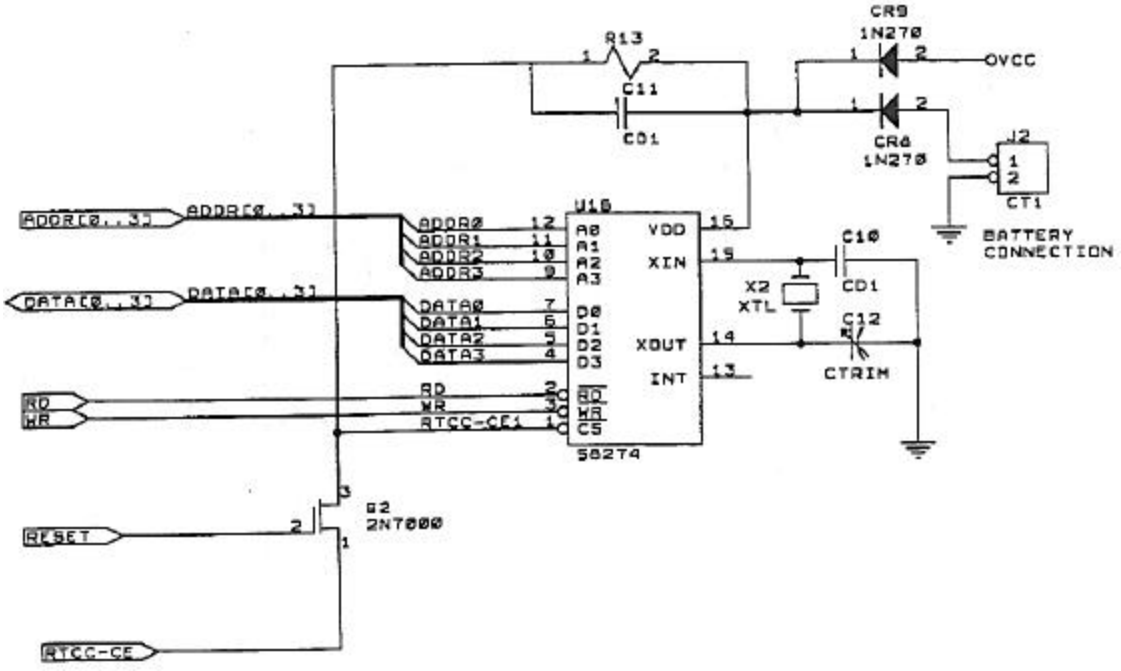
5. Lines 8523-8610: The INPutted year is reduced to a digit (0-3) to determine the number of years that the inputted date in D$ exceeds a leap year. The result is stored in register 15. Note the use of logical **AND (&)** in line 8528 to mask other date in Register 15.

TEST PROGRAM - REAL-TIME CALENDAR CLOCK

```
7002    PRI:PRI:PRI"TIME-DATE MENU:"
7004    PRI" <1> READ TIME <3> SET TIME"
7005    PRI" <2> READ DATE <4> SET DATE"
7010    PRI:PRI"ENTER SELECTION:";:INPUT Z0
7020    IF Z0=1 THEN GOTO 8686
7021    IF Z0=2 THEN GOTO 8650
7022    IF Z0=3 THEN GOTO 8670
7023    IF Z0=4 THEN GOTO 8505
8505    PRI:PRI"TYPE DATE (MM/DD/YY):":INPUT D$
8510    IF LEN(D$) > 8 THEN GOTO 8600
8512    IF LEN(D$) < 8 THEN GOTO 8600
8514    PRI:PRI"ENTER DAY-OF-WEEK (1=SUN 7=SAT): ";:INPUT D0
8516    OUT 160,1
8518    OUT 171, VAL(LEFT$(D$,1)):OUT 170, VAL(MID$(D$,2,1))
8519    OUT 169, VAL(MID$(D$,4,1)):OUT 168, VAL(MID$(D$,5,1))
8520    OUT 173, VAL(MID$(D$,7,1)):OUT 172, VAL(MID$(D$,8,1)):OUT 174,D0
8523    D2=VAL(MID$(D$,7,2))-84
8527    IF D2 > 4 THEN GOTO 8610
8528    D3=((INP(175)-240) & 2)
8529    OUT 175,(D2*4+D3):GOTO 7002
8600    PRI"INCORRECT FORMAT":GOTO 8505
8610    D2=D2-4:GOTO 8527
8650    D4=INP(171)-192:D5=INP(170)-192:D6=INP(169)-192:D7=INP(168)-192
8652    D8=INP(173)-192:D9=INP(172)-192:D0=INP(174)-240
8654    D3$="/"
8655    D1$=CHR$(D4)+CHR$(D5)+D3$+CHR$(D6)+CHR$(D7)+D3$+CHR$(D8)+CHR$(D9)
8657    IF D0=1 THEN D2$=" - SUN"
8658    IF D0=2 THEN D2$=" - MON"
8659    IF D0=3 THEN D2$=" - TUE"
8660    IF D0=4 THEN D2$=" - WED"
8662    IF D0=5 THEN D2$=" - THU"
```

```
8663    IF D0=6 THEN D2$=" - FRI"
8664    IF D0=7 THEN D2$=" - SAT"
8668    PRI:PRI" >>>> DATE IS:";:PRI D1$+D2$;:PRI" <<<<":GOTO 7002
8670    PRI:PRI"ENTER TIME:":PRI"FORMAT IS HH:MM-AM OR HH:MM-PM (8 CHAR)"
8672    PRI"`RETURN' WHEN SET TIME=ACTUAL TIME"
8673    PRI:PRI"TIME IS:";:INPUT T$
8674    IF MID$(T$,7,1)="A" THEN T0=0
8675    IF MID$(T$,7,1)="P" THEN T0=2
8676    T1=INP(175):T2=(T1 & 12)+T0
8680    OUT 167, VAL(LEFT$(T$,1)):OUT 166,VAL(MID$(T$,2,1)):OUT 165,VAL(MID$(T$,4,1))
8682    OUT 164, VAL(MID$(T$,5,1)):OUT 162,0:OUT 163,0:OUT 175, T2
8684    GOTO 7002
8686    T1=INP(167)-192:T2=INP(166)-192:T3=INP(165)-192:T4=INP(164)-192
8687    T5=INP(163)-192:T6=INP(162)-192:T7=(INP(175) & 2)
8689    IF T7=0 THEN T3$=" AM"
8690    IF T7=2 THEN T3$=" PM"
8692    T1$=CHR$(T1)+CHR$(T2)+CHR$(58)+CHR$(T3)+CHR$(T4)+CHR$(58)
8693    T2$=(CHR$(T5)+CHR$(T6))
8696    PRI:PRI" >>>> TIME IS:";:PRI T1$+T2$+T3$;:PRI" <<<<":GOTO 7002
```

SCHEMATIC DIAGRAM



National Semiconductor

MM58274 Microprocessor Compatible Real Time Clock

General Description

The MM58274 is fabricated using low threshold metal gate
CMOS technology and is designed to operate in bus Oriented microprocessor systems where a real time clock
And calendar function are required. The on-chip 32.768 KHz crystal controlled oscillator will maintain timekeeping
Down to 2.2V to allow low power standby battery operation.
This device is pin compatible with the MM58174A but Continues timekeeping up to tens of years. Faster access
Times are also offered.

Applications

- Point of sale terminals
- Teller terminals
- Word processors
- Data logging
- Industrial process control

Features

- Same pin-out as MM58174A
- Timekeeping from tenths of seconds to tens of years in independently accessible registers
- Leap year register
- Hours counter programmable for 12 or 24-hour operation
- Buffered crystal frequency output in test mode for easy oscillator setting
- Data-changed flag allows simple testing for time rollover
- Independent interrupting timer with open drain output
- Fully TTL compatible
- Low power standby operation (10 A at 2.2V)
- Low cost 16-pin DIP

Block Diagram

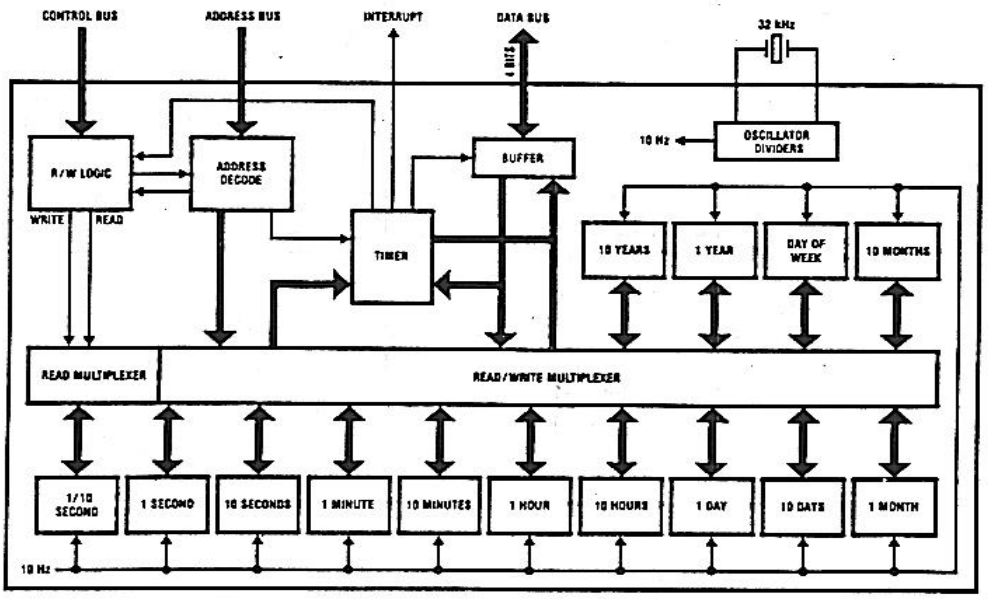


FIGURE 1

Jameco Part number 148726

MM58274

Absolute Maximum Ratings (Note 1)

DC Input or Output Voltage	-0.3V to V_{DD}+0.3V
DC Input or Output Diode Current	± 5.0mA
Storage Temperature, T_{STG}	-65°C to + 150°C
Supply Voltage V_{DD}	6.5V
Power Dissipation, P_D	500 mW
Lead Temperature(Soldering, 10 seconds)	260 °C

Operating Conditions

	Min	Max	Unit
Operating Supply Voltage	4.5	5.5	s
Standby Mode Supply Voltage	2.2	5.5	V
DC Input or Output Voltage	0	V_{DD}	V
Operating Temperature Range	-40	85	V °C

Electrical Characteristics V_{DD}=5V ± 10%, T = -40°C to +85°C unless otherwise stated

Symbol	Parameter	Conditions	Min	Typ	Max	Units
V_{IH}	High Level Input Voltage (except XTAL IN)		2.0			V
V_{IL}	Low Level Input Voltage (except XTAL IN)		-		0.8	V
V_{OH}	High Level Output Voltage (DB0-DB3)	I_{OH} = -20 ìA I_{OH} = -1.6 mA	V_{DD} - 0.1 3.7			V V
V_{OH}	High Level Output Voltage (INT)	I_{OH} = -20 ìA (In Test Mode)	V_{DD} - 0.1			V
V_{OL}	Low Level Output Voltage (DB0-DB3, INT)	I_{OL} = 20 ìA I_{OL} = 1.6 mA			0.1 0.4	V V
I_{IL}	Low Level Input Current (AD0-AD3, DB0-DB3)	V_{IN} = V_{SS} (Note 2)			-80	ìA
I_{IL}	Low Level Input Current (WR, RD)	V_{IN} = V_{SS} (Note 2)			-190	ìA
I_{IL}	Low Level Input Current (CS)	V_{IN} = V_{SS} (Note 2)			-550	ìA
I_{OZH}	Output High Level Leakage Current (INT)	V_{OUT} = V_{DD}			2.0	ìA ìA
I_{DD}	Average Supply Current	V_{DD} = 2.2V (Standby Mode) V_{DD} = 5.0V (Active Mode)		4	10 1	ìA mA
C_{IN}	Input Capacitance			5	10	pF
C_{OUT}	Output Capacitance	(Outputs Disabled)		10		pF

Note 1: Absolute Maximum Ratings are those values beyond which damage to the device may occur. All voltages referenced to ground unless otherwise noted.

Note 2: The DB0-DB3 and AD0-AD3 lines all have active P-channel pull-up transistors which will source current. The CS, RD, and WR lines have internal pullup resistors to V_{DD}.

AC Switching Characteristics

READ TIMING: DATA FROM PERIPHERAL TO MICROPROCESSOR V_{OO} = 5V ± 0.5V, C_L = 100 pF

Symbol	Parameter	Commercial Specification T_A = = 40°C to + 85°C			Units
		Min	Typ	Max	
t_{AD}	Address Bus Valid to Data Valid		500	850	ns
t_{CSD}	Chip Select On to Data Valid		250	425	ns
t_{RD}	Read Strobe On to Data Valid		250	425	ns
t_{RW}	Read Strobe Width (Note 3)			DC	
t_{RA}	Address Bus Hold Time from Trailing Edge of Read Strobe	0			ns
t_{CSH}	Chip Select Hold Time from Trailing Edge of Read Strobe	0			ns
t_{RH}	Data Hold Time from Trailing Edge of Read Strobe	100	250		ns
t_{HZ}	Time from Trailing Edge of Read Strobe Unit O/P Drivers are TRI-STATE	200	250	340	ns

WRITE TIMING: DATA FROM MICROPROCESSOR TO PERIPHERAL V_{DD} = 5V ± 0.5V.

Symbol	Parameter	Commercial Specification T_A = = 40°C to + 85°C			Units
		Min	Typ	Max	
t_{AW}	Address Bus Valid to Write Strob (Note 4)	600			ns
t_{CSW}	Chip Select On to Write Strob	350	175		ns
t_{DW}	Data Bus Valid to Write Strobe	600	400		ns
t_{WW}	Write Strobe Width	350			ns
t_{WCS}	Chip Select Hold Time Following Write Strobe	0			ns
t_{WA}	Address Bus Hold Time Following Write Strobe	0			ns
t_{WD}	Data Bus Hold Time Following Write Strobe	200			ns

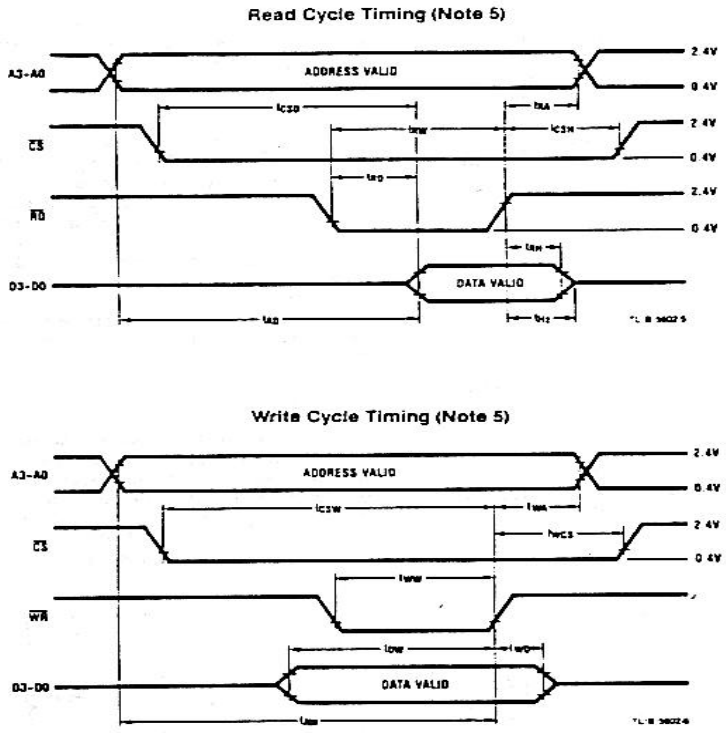
Note 3: Except for special case restriction: with interrupts programmed, max read strobe width of control register (ADDR 0) is 30 ms. See section on interrupt Programming.

Note 4: All timings measured to the trailing edge of write strobe (data latched by the trailing edge of $\overline{WR}$).

Note 5: Input test waveform peak voltages are 2.4V and 0.4V. Output signals are measured to their 2.4V and 0.4V levels.

Jameco Part number 148726

Switching Time Waveforms



Connection Diagram

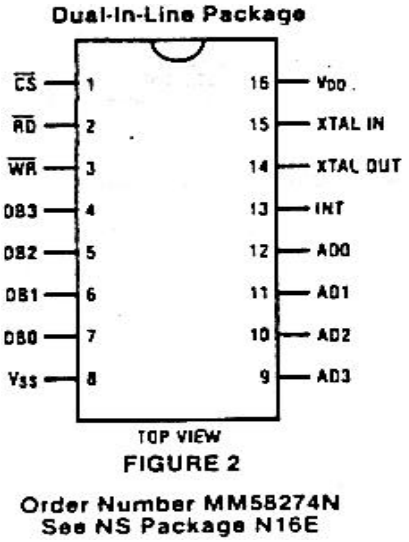


FIGURE 2

Order Number MM58274N
See NS Package N16E

Functional Description

The MM58274 is a bus oriented microprocessor real time clock. It has the same pin-out as the MM58174A while offering extended timekeeping up to units and tens of years. To enhance the device further, a number of other features have been added including: 12 or 24 hours counting, a testable data-changed flag giving easy error-free time reading and simplified interrupt control.

A buffered oscillator signal appears on the interrupt output when the device is in test mode. This allows for easy oscillator setting when the device is initially powered up in a system.

The counters are arranged as 4-bit-words and can be randomly accessed for time reading and setting. The counters output in BCD (binary coded decimal) 4-bit numbers. Any register which has less than 4 bits (e.g., days of week uses only 3 bits) will return a logic 0 on any unused bits. When written to, the unused inputs will be ignored.

Jameco Part number 148726

Functional Description

Writing a logic 1 to the clock start/stop control bit resets the internal osciliator divider chain and the tenths of seconds counter. Writing a logic 0 will start the clock timing from the nearest second. The time then updates every 100 ms with all counters changing synchronously. Time changing during a read is detected by testing the data-changed bit of the control register after completing a string of clock register reads.

Interrupt delay times of 0, 1s, 0.5s, 1s, 5s, 10s, 30s or 60s can be selected with single or repeated interrupt outputs. The open drain output is pulled low whenever the interrupt timer times out and is cleared by reading the control register.

CIRCUIT DESCRIPTION

The block diagram in *Figure 1* shows the internal structure of the chip. The 16-pin package outline is shown in *Figure 2*.

Crystal Oscillator

This consists of a CMOS inverter/amplifier with an on-chip bias resistor. Externally a 20 pF capacitor, a 6 pF-36 pF trimmer capacitor and a crystal are required to complete the 32.768 kHz timekeeping oscillator circuit.

The 6 pF-36 pF trimmer fine tunes the crystal load impedance, optimizing the osciliator stability. When properly adjusted (i.e., to the crystal frequency of 32.768 kHz), the circuit will display a frequency variation with voltage of less than 3 ppm/V.

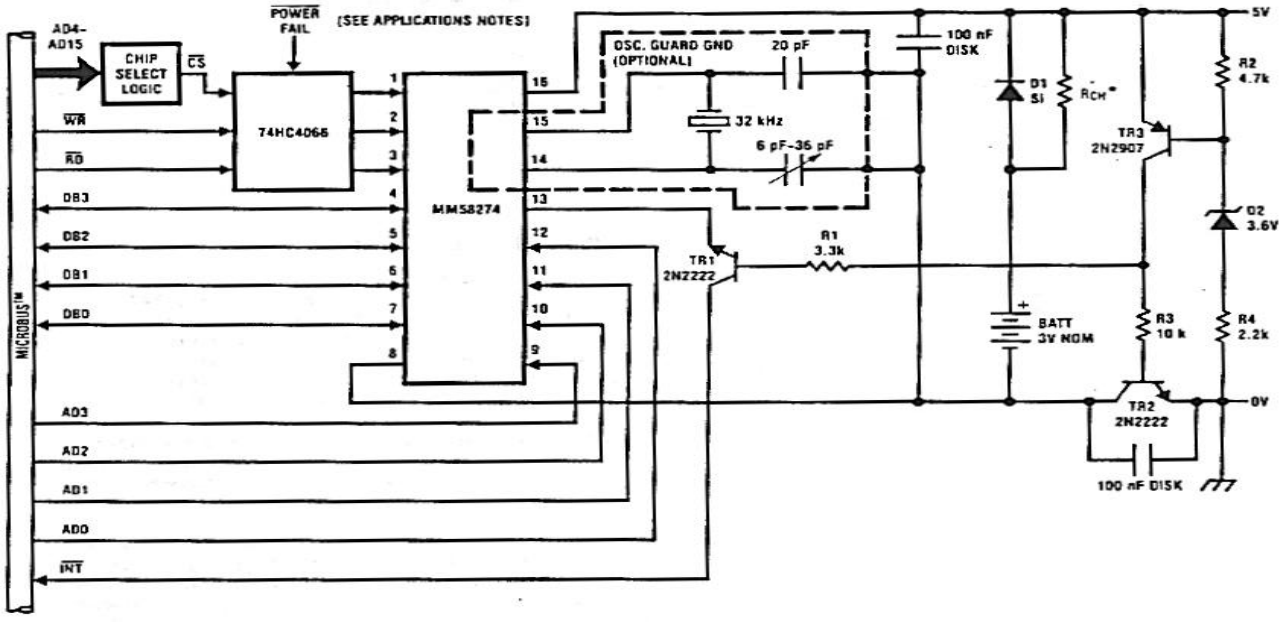
When the chip is enabled into test mode, the oscillator is gated onto the interrupt output pin giving a buffered oscillator output that can be used to set the crystal frequency when the device is installed in a system. For further information see the section on Test Mode.

Divider Chain

The crystal oscillator is divided down in three stages to produce a 10 Hz frequency setting pulse. The first stage is a non-integer divider which reduces the 32.768 kHz input to 30.720 kHz. This is further divided by a 9-stage binary ripple counter giving an output frequency of 60 Hz. A 3-stage Johnson counter divides this by six, generating a 10 Hz output. The 10 Hz clock is gated with the 32.768 kHz crystal frequency to provide clock setting pulses of 15.26 μs duration. The setting pulse drives all the time registers on the device which are synchronously clocked by this signal. All time data and the data-changed flag change on the failing edge of the clock setting pulse.

Data-Changed Flag

The data-changed flag is set by the clock setting pulse to indicate that the time data has been altered since the clock was last read. This flag occupies bit 3 of the control register where it can be tested by the processor to sense data-changed. It will be reset by a read of the control register. See the section, "Methods of Device Operation", for suggested clock reading techniques using this flag.



* Resistor is only used with Ni-CAD cells. Omit for lithium, silver or other primary cells.

FIGURE 3. Typical System Connection Diagram

Jameco Part number 148726
Functional Description

Seconds Counters

There are three counters for seconds:

a) tenths of seconds

b) units of seconds

c) tens of seconds.

The registers are accessed at the addresses shown in Table 1. The tenths of seconds register is reset to 0 when the clock start/stop bit (bit 2 of the control register) is set to logic 1. The units and tens of seconds are set up by the processor, giving time setting to the nearest second. All three registers can be read by the processor for time output.

Minutes Counters

There are two minutes counters:

a) units of minutes

b) tens of minutes.

Both registers may be read to or written from as required.

Hours Counters

There are two hours counters:

a) units of hours

b) tens of hours.

Both counters may be accessed for read or write operations as desired.

In 12-hour mode, the tens of hours register has only one active bit and the top three bits are set to logic 0. Data bit 1 of the clock setting register is the AM/PM indicator; logic 0 indicating AM, logic 1 for PM.

When 24-hour mode is programmed, the tens of hours register reads out two bits of data and the two most significant bits are set to logic 0. There is no AM/PM indication and bit 1 of the clock setting register will read out a logic 0.

In both 12/24-hour modes, the units of hours will read out four active data bits, 12 or 24-hour mode is selected by bit 0 of the clock setting register, logic 0 for 12-hour mode, logic 1 for 24-hour mode.

Days Counters

There are two days counters:

a) units of days

b) tens of days.

The days counters will count up to 28, 29, 30 or 31 depending on the state of the months counters and the leap year counter. The microprocessor has full read/write access to these registers.

Months Counters

There are two months counters:

a) units of months

b) tens of months.

Both these counters have full read/write access.

Years Counters

There are two years counters:

a) units of years

b) tens of years.

Both these counters have full read/write access. The years will count up to 99 and roll over to 00.

TABLE 1. Address Decoding of Real-Time Clock Internal Registers

Register Selected	Address (Binary)				(Hex)	Access
	AD3	AD2	AD1	AD0		
0 Control Register	0	0	0	0	0	Split Read and Write
1 Tenths of Seconds	0	0	0	1	1	Read Only
2 Units Seconds	0	0	1	0	2	R/W
3 Tens Seconds	0	0	1	1	3	R/W
4 Units Minutes	0	1	0	0	4	R/W
5 Tens Minutes	0	1	0	1	5	R/W
6 Units Hours	0	1	1	0	6	R/W
7 Tens Hours	0	1	1	1	7	R/W
8 Units Days	1	0	0	0	8	R/W
9 Tens Days	1	0	0	1	9	R/W
10 Units Months	1	0	1	0	A	R/W
11 Tens Months	1	0	1	1	B	R/W
12 Units Years	1	1	0	0	C	R/W
13 Tens Years	1	1	0	1	D	R/W
14 Day of Week	1	1	1	0	E	R/W
15 Clock Setting/Interrupt Registers	1	1	1	1	F	R/W

Jameco Part number 148726

Functional Description

Day of Week Counter

The day of week counter increments as the time rolls from 23:59 to 00:00 (11:59 PM to 12:00 AM in 12-hour mode). It counts from 1 to 7 and rolls back to 1. Any day of the week may be specified as day 1.

Clock Setting Register/Interrupt Register

The interrupt select bit in the control register determines which of these two registers is accessible to the processor at address 15. Normal clock and interrupt timing operations will always continue regardless of which register is selected onto the bus. The layout of these registers is shown in Table II.

The clock setting register is comprised of three separate functions:

a) leap year counter: bits 2 and 3

b) AM/PM indicator: bit 1

c) 12/24-hour mode set: bit 0 (see Table 11A).

The leap year counter is a 2-stage binary counter which is clocked by the months counter. It changes state as the time rolls over from 11:59 on December 31 to 00:00 on January 1.

The counter should be loaded with the `number of years since last leap year' e.g., if 1980 was the last leap year, a clock programmed in 1983 should have 3 stored in the leap year counter. If the clock is programmed during a leap year, then the leap year counter should be set to 0. The contents of the leap year counter can be read by the P.

The AM/PM indicator returns a logic 0 for AM and a logic 1 for PM. It is clocked when the hours counter rolls from 11:59 to 12:00 in 12-hour mode. In 24-hour mode this bit is set to logic 0.

The 12/24-hour mode set determines whether the hours counter counts from 1 to 12 or from 0 to 23. It also controls the AM/PM indicator, enabling it for 12-hour mode and forcing it to logic 0 for the 24-hour mode. The 12/24-hour mode bit is set to logic 0 for 12-hour mode and it is set to logic 1 for 24-hour mode.

IMPORTANT NOTE: *Hours mode and AM/PM bits cannot be set in the same write operation. See the section on* Initialization (Methods of Device Operation) *for a suggested setting routine*.

All bits in the clock setting register may be read by the processor.

The interrupt register controls the operation of the timer for interrupt output. The processor programs this register for single or repeated interrupts at the selected time intervals.

The lower three bits of this register set the time delay period that will occur between interrupts. The time delays that can be programmed and the data words that select these are outlined in Table 11B.

Data bit 3 of the interrupt register sets for either single or repeated interrupts; logic 0 gives single mode, logic 1 sets for repeated mode.

Using the interrupt is described in the Device Operation section.

TABLE 11A. Clock Setting Register Layout

Function	Data Bits Used DB3	DB2	DB1	DB0	Comments	Access
Leap Year Counter	**X**	**X**			0 Indicates a Leap Year	R/W
AM/PM Indicator (12-Hour Mode)			X		0 = AM 1 = PM 0 in 24-Hour Mode	R/W
12/24-Hour Select Bit				X	0 = 12 Hour Mode 1 = 24-Hour Mode	R/W

TABLE 11B. Interrupt Control Register

Function	Comments	Control Word DB3	DB2	DB1	DB0
No interrupt	Interrupt output cleared, start/stop bit set to 1.	X	0	0	0
0.1 Second		0/1	0	0	1
0.5 Second		0/1	0	1	0
1 Second	DB3=0 for single interrupt	0/1	0	1	1
5 Seconds	DB3 = 1 for repeated interrupt	0/1	1	0	0
10 Seconds		0/1	1	0	1
30 Seconds		0/1	1	1	0
60 Seconds		0/1	1	1	1

Timing Accuracy: single interrupt mode (all time delays): ± 1 ms
Repeated Mode: ± 1 ms on initial timeout, thereafter synchronous with first interrupt (i.e., timing errors do not accumulate).

Jameco Part number 148726

Functional Description

Control Register

There are three registers which control different operations of the clock:

a) the clock setting register

b) the interrupt register

c) the control register.

The clock setting and interrupt registers both reside at address 15, access to one or the other being controlled by the interrupt select bit; data bit 1 of the control register.

The clock setting register programs the timekeeping of the clock. The 12/24-hour mode select and the AM/PM indicator for 12-hour mode occupy bits 0 and 1, respectively. Data bits 2 and 3 set the leap year counter.

The interrupt register controls the operation of the interrupt timer, selecting the required delay period and either single or repeated interrupt.

The control register is responsible for controlling the operations of the clock and supplying status information to the processor. It appears as two different registers; one with write only access and one with read only access.

The write only register consists of a bank of four latches which control the internal processes of the clock.

The read only register contains two output data latches which will supply status information for the processor. Table III shows the mapping of the various control latches and status flags in the control register. The control register is located at address 0.

The write only portion of the control register contains four latches:

A logic 1 written into the test bit puts the device into test mode. This allows setting of the oscillator frequency as well as rapid testing of the device registers, if required. A more complete description is given in the Test Mode section. For normal operation the test bit is loaded with logic 0.

The clock start/stop bit stops the timekeeping of the clock and resets to 0 the tenths of seconds counter. The time of day may then be written into the various clock registers and the clock restarted synchronously with an external time source. Timekeeping is maintained thereafter.

A logic 1 written to the start/stop bit halts clock timing. Timing is restarted when the start/stop bit is written with a logic 0.

The interrupt select bit determines which of the two registers mapped onto address 15 will be accessed when this address is selected.

A logic 0 in the interrupt select bit makes the clock setting register available to the processor. A logic 1 selects the interrupt register.

The interrupt start/stop bit controls the running of the interrupt timer. It is programmed in the same way as the clock start/stop bit; logic 1 to halt the interrupt and reset the timer, logic 0 to start interrupt timing.

When no interrupt is programmed (interrupt control register set to 0), the interrupt start/stop bit is automatically set to a logic 1. When any new interrupt is subsequently programmed, timing will not commence until the start/stop bit is loaded with 0.

In the single interrupt mode, interrupt timing stops when a timeout occurs. The processor restarts timing by writing logic 0 into the start/stop bit.

In repeated interrupt mode the interrupt timer continues to count with no intervention by the processor necessary.

Interrupt timing may be stopped in either mode by writing a logic 1 into the interrupt start/stop bit. The timer is reset and can be restarted in the normal way, giving a full time delay period before the next interrupt.

In general, the control register is set up such that writing 0's into it will start anything that is stopped, pull the clock out of test mode and select the clock setting register onto the bus. In other words, writing 0 will maintain normal clock operation and restart interrupt timing, etc.

The read only portion of the control register has two status outputs:

Since the MM58274 keeps real time, the time data changes asynchronously with the processor and this may occur while the processor is reading time data out of the clock.

Some method of warning the processor when the time data has changed must thus be included. This is provided for by the data-changed flag located in bit 3 of the control register. This flag is set by the clock setting pulse which also clocks the time registers. Testing this bit can tell the processor whether or not the time has changed. The flag is cleared by a read of the control register but not by any write operations. No other register read has any effect on the state of the data-changed flag.

Data bit 0 is the interrupt flag. This flag is set whenever the interrupt timer times out, pulling the interrupt output low. in a polled interrupt routine the processor can test this flag to determine if the MM58274 was the interrupting device. This interrupt flag and the interrupt output are both cleared by a read of the control register.

TABLE III. The Control Register Layout

Access (addr0)	DB3	DB2	DB1	DB0
Read From:	Data-Changed Flag	0	0	Interrupt Flag
Write To:	Test 0=Normal 1=Test Mode	Clock Start/Stop 0=Clock Run 1=Clock Stop	Interrupt Select 0=Clock Setting Register 1=Interrupt Register	Interrupt Start/Stop 0=Interrupt Run 1=Interrupt Stop

Functional Description

Both of the flags and the interrupt output are reset by the trailing edge of the read strobe. The flag information is held latched during a control register read, guaranteeing that stable status information will always be read out by the processor.

Interrupt timeout is detected and stored internally if it occurs during a read of the control register, the interrupt output will then go low only after the read has been completed.

A clock setting pulse occurring during a control register read will not affect the data-changed flag since time data read out before or after the control read will not be affected by the time change.

METHODS OF DEVICE OPERATION
Test Mode

National Semiconductor uses test mode for functionally testing the MM58274 after fabrication and again after packaging. Test mode can also be used to set up the oscillator frequency when the part is first commissioned.

Figure 4 shows the internal clock connections when the device is written into test mode. The 32.768 kHz oscillator is gated onto the interrupt output to provide a buffered output for initial frequency setting. This signal is driven from a TRI-STATE output buffer, enabling easy oscillator setting in systems where interrupt is not normally used and there is no external resistor on the pin.

If an interrupt is programmed, the 32.768 kHz output is switched off to allow high speed testing of the interrupt timer. The interrupt output will then function as normal.

The clock start/stop bit can be used to control the fast clocking of the time registers as shown in *Figure 4*.

MM58274 Initialization

When it is first installed and power is applied, the device will need to be properly initialized. The following operation steps are recommended when the device is set up (all numbers are decimal):

1) Disable interrupt on the processor to allow oscillator setting. Write 15 into the control register: *The clock and interrupt start/stop bits are set to 1. ensuring that the clock and interrupt timers are both halted. Test mode and the interrupt register are selected.*

2) Write 0 to the interrupt register: *Ensure that there are no interrupts programmed and that the oscillator will be gated onto the interrupt output.*

3) Set oscillator frequency: *All timing has been halted and the oscillator is buffered out onto the interrupt line.*

4) Write 5 to the control register: *The clock is now out of test mode but is still halted. The clock setting register is now selected by the interrupt select bit.*

5) Set 12/24 Hours Mode: *Write to the clock setting register to select the hours counting mode required.*

6) Load Real-Time Registers: *All time registers (including Leap Years and AM/PM bit) may now be loaded in any order. Note that when writing to the clock setting register to set up Leap Years and AM/PM, the Hours Mode bit must not be altered from the value programmed in step 5.*

7) Write 0 to the control register: *This operation finishes the clock initialization by starting the time. The final control register write should be synchronized with an external time source.*

In general, timekeeping should be halted before the time data is altered in the clock. The data can, however, be altered at any time if so desired. Such may be the case if the user wishes to keep the clock corrected without having to stop and restart it; i.e., winter/summer time changing can be accomplished without halting the clock. This can be done in software by sensing the state of the datachanged flag and only altering time data just after the time has rolled over (data-changed flag set).

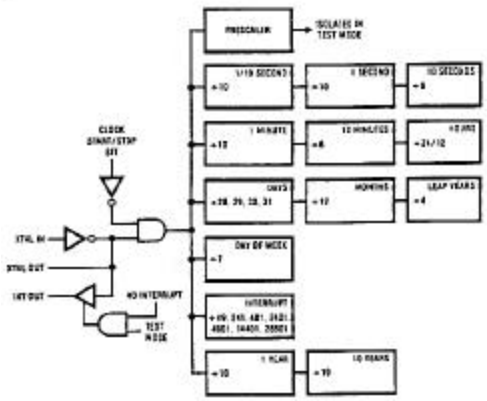


FIGURE 4. Test Mode Organization

Jameco Part number 148726

Functional Description

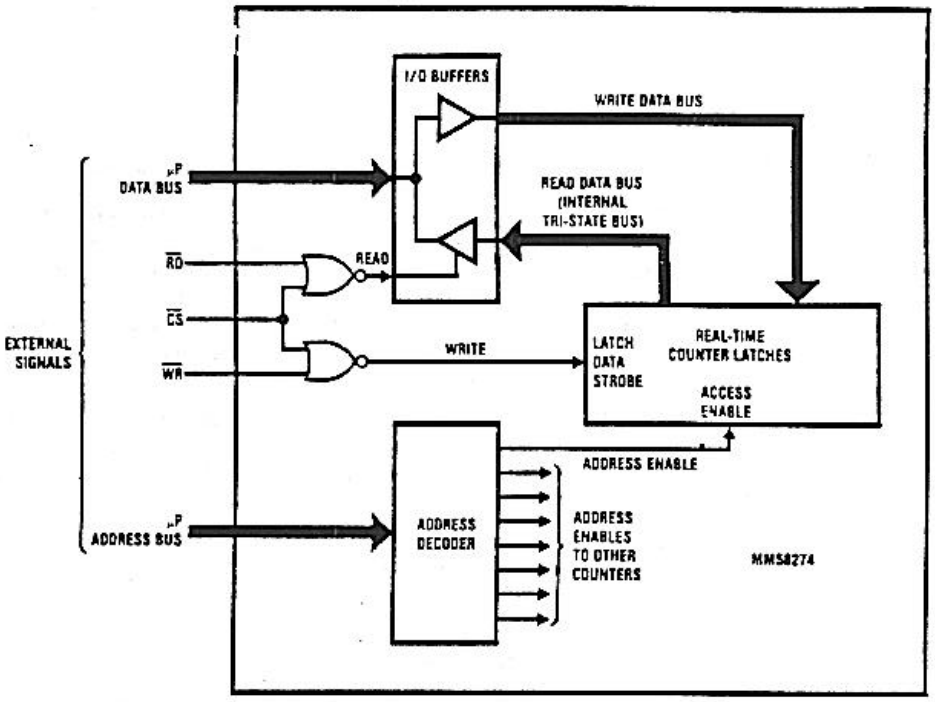


FIGURE 5. MM58274 Microprocessor Interface Diagram

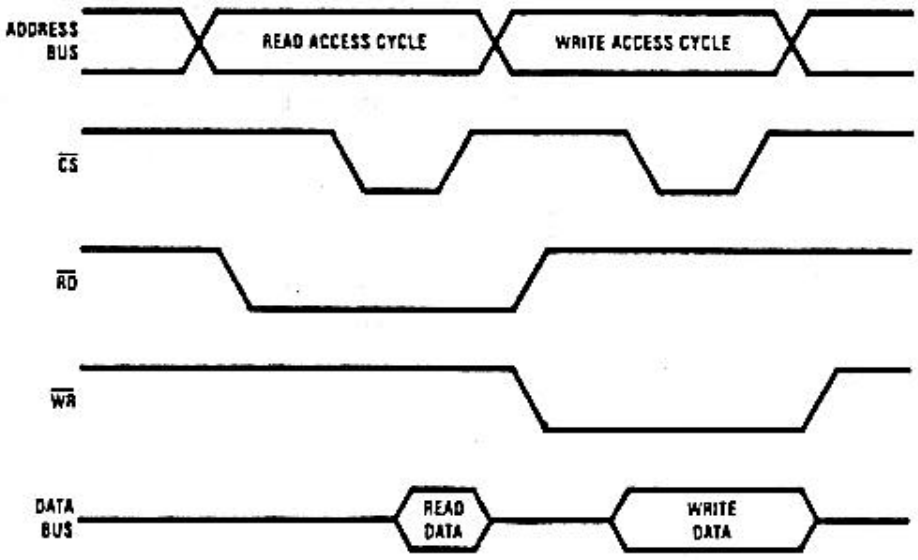


FIGURE 6. Valid MM58274 Control Signals Using Chip Select Generated Access Strobes

APPLICATION NOTES

Time Reading Using Interrupt

In systems such as point of sale terminals and data loggers, time reading is usually only required on a random demand basis. Using the data-changed flag as outlined in the section on methods of operation is ideal for this type of system. Some systems, however, need to sense a change in real time: e.g., industrial timers/process controllers, TV/VCR clocks, any system where real time is displayed.

The interrupt timer on the MM58274 can generate interrupts synchronously with the time registers changing, using software to provide the initial synchronization.

In single interrupt mode the processor is responsible for initiating each timing cycle and the timed period is accurate to ±1 ms.

In repeated interrupt mode the period from the initial processor start to the first timeout is also only accurate to ±1ms. The following interrupts maintain accurate delay periods relative to the first timeout. Thus, to utilize interrupt to control time reading, we will use repeated interrupt mode.

In repeated mode the time period between interrupts is exact, which means that timeouts will always occur at the same point relative to the internal clock setting pulses. The case for 0.1s interrupts is shown in *Figure A.1*. The same is true for other delay periods, only there will be more clock setting pulses between each interrupt timeout. If we set up the interrupt timer so that interrupt always times out just after the clock setting pulse occurs (*Figure A.2*), then there is no need to test the data-changed flag as we know that the time data has just changed and will not alter again for another 100 ms.

This can be achieved as outlined below:

1) Follow steps 1 and 2 of the section on interrupt programming. In step 2 set up for repeated interrupt.

2) Read control register ADO: *This is a dummy read to reset the data-changed flag*.

3) Read control register ADO until data-changed flag is set.

4) Write 0 or 2 to control register. Interrupt timing commences.

Time Reading with Very Slow Read Cycles

If a system takes longer than 100 ms to complete reading of all the necessary time registers (e.g., when CMOS processors are used) or where high level interpreted language routines are used, then the data-changed flag will always be set when tested and is of no value. In this case, the time registers themselves must be tested to ensure data accuracy.

The technique below will detect both time changing between read strobes (i.e., between reading tens of minutes and units of hours) and also time changing *during* read, which can produce invalid data.

1) Read and store the value of the lowest order time register required.

2) Read out all the time registers required. The registers may be read out in any order, simplifying software requirements.

3) Read the lowest order register and compare it with the value stored previously in step 1. If it is still the same, then all time data is good. If it has changed, then store the new value and go back to step 2.

In general, the rule is that the first and last reads *must* both be of the lowest order time register. These two values can then be compared to ensure that no change has occurred. This technique works because for any higher order time register to change, all the lower order registers must also change. If the lowest order register does not change, then no higher order register has changed either.

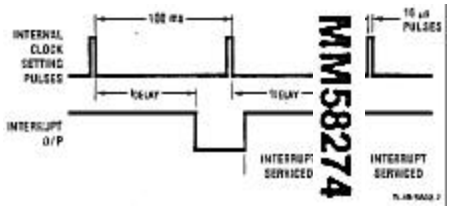


FIGURE A.1. Time Delay from Clock Setting Pulses to Interrupt is Constant

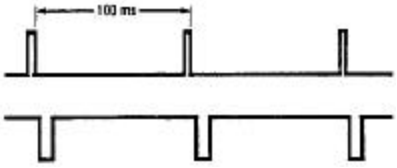


FIGURE A.2. Interrupt Timer Synchronized with Clock Setting Pulses

Jameco Part number 148726

Functional Description

Reading the Time Registers

Using the data-changed flag technique supports microprocessors with block move facilities, as all the necessary time data may be read sequentially and then tested for validity as shown below.

1) Read the control register, address 0: *This is a dummy read to reset the data-changed flag (DCF) prior to reading the time registers*.

2) Read time registers: *All desired time registers are read out in a block*.

3) Read the control register and test DCF: *If DCF is cleared (logic 0), then no clock setting pulses have occurred since step 1. All time data is guaranteed good and time reading is complete*.

If DCF is set (logic 1), then a time change has occurred since step 1 and time data may not be consistent. Repeat steps 2 and 3 until DCF is clear. The control read of step 3 will have reset DCF, automatically repeating the step 1 action.

Interrupt Programming

The interrupt timer generates interrupts at time intervals which are programmed into the interrupt register. A single interrupt after delay or repeated interrupts may be programmed. Table IIB lists the different time delays and the data words that select them in the interrupt register.

Once the interrupt register has been used to set up the delay time and to select for single or repeat, it takes no further part in the workings of the interrupt system. All activity by the processor then takes place in the control register.

Initializing:

1) Write 3 to the control register (ADO): *Clock timing continues, interrupt register selected and interrupt timing stopped*.

2) Write interrupt control word to address 15: *The interrupt register is loaded with the correct word (chosen from Table IIB) for the time delay required and for single or repeated interrupts*.

3) Write 0 or 2 to the control register: *Interrupt timing commences. Writing 0 selects the clock setting register onto the data bus; writing 2 leaves the interrupt register selected. Normal timekeeping remains unaffected*.

On Interrupt:

Read the control register and test for Interrupt Flag (bit 0).

If the flag is cleared (logic 0), then the device is not the source of the interrupt.

If the flag is set (logic 1), then the clock did generate an interrupt. The flag is reset and the interrupt output is cleared by the control register read that was used to test for interrupt.

Single Interrupt Mode:

When appropriate, write 0 or 2 to the control register to restart the interrupt timer.

Repeated Interrupt Mode:

Timing continues, synchronized with the control register write which originally started interrupt timing. No further intervention is necessary from the processor to maintain timing.

In either mode interrupt timing can be stopped by writing 1 into the control register (interrupt start/stop set to 1). Timing for the full delay period recommences when the interrupt start/stop bit is again loaded with 0 as normal.

IMPORTANT NOTE: Using the interrupt timer places a constraint on the maximum Read Strobe width which may be applied to the clock. Normally all registers may be read from with a tRW down to DC (i.e., CS and RD held continuously low). When the interrupt timer is active however, the maximum read strobe width that can be applied to the control register (Addr 0) is 30 ms.

This restriction is to allow the interrupt timer to properly reset when it times out. Note that it only affects reading of the control register--all other addresses in the clock may be accessed with DC read strobes, regardless of the state of the interrupt timer. Writes to any address are unaffected.

NOTES ON AC TIMING REQUIREMENTS

Although the Switching Time Waveforms show Microbus control signals used for clock access, this does not preclude the use of the MM58274 in other non-Microbus systems. *Figure 5* is a simplified logic diagram showing how the control signals are gated internally to control access to the clock registers. From this diagram it is clear that CS could be used to generate the internal data transfer strobes, with RD and WR input set up first. This situation is illustrated in Figure 6.

The internal data busses of the MM58274 are fully CMOS, contributing to the flexibility of the control inputs. When determining the suitability of any given control signal pattern for the MM58274, the timing specifications in AC Switching Characteristics should be examined. As long as these timings are met (or exceeded) the MM58274 will function correctly.

When the MM58274 is connected to the system via a peripheral port, the freedom from timing constraints allows for very simple control signal generation, as in Figure 7. For reading (*Figure 7a*), Address, CS and RD may be activated simultaneously and the data will be available at the port after t_{AD} max (850 ns). For writing (*Figure 7b*), the address and data may be applied simultaneously and CS and WR strobed together.